Model Reduction using Domain Decomposition Long Short Term Memory Neural Networks

Project Plan for ACSE-9

Author: Yunzhen (Wade) Song
Supervisors: Christopher Pain, Claire Heaney

June 28, 2019

1 Motivation

Reduced order modelling is a powerful technique for rapidly modelling high dimensional fluid dynamics systems. Its speed could enable real-time decision making and operational modelling. The Reduced Order Model (ROM) can exploit Domain Decomposition (DD) methods in order to increase its accuracy and further reduced the computational time for a large system[27;26;25], see Figure 1&2 .

2 Project Objectives

Long Short Term Memory (LSTM) neural networks have demonstrated high accuracy on time-dependent deterministic problems. They have been widely adopted in stocks price prediction, driverless cars, chess playing, translation and speech recognition. The 'memory' possessed by these neural networks is particularly important for model reduction and will be used for large scale problems. For example, these methods may be able to resolve the flows within a building while simultaneously resolving the flows within an entire city. This project will develop LSTM networks by combining them with Domain Decomposition methods to produce a new Domain Decomposition Long Short Term Memory Neural Network DD-LSTM. This has an increase recurency over the 'standard' LSTM as one needs to iterate between the sub-domains in order to distribute the sub-domain solutions across the whole domain. This is important in order to satisfy the impressibility constrain in many fluid flow problems.

The project will be divided into three steps:

The first step is using an existing High Fidelity Model (HFM) based on discretizations of the governing equations, such as using the finite element method for solving the advection equation or the Navier-Stokes equations that govern fluid flows. The data comes from the snapshots taken at different time levels of the HFM. Dimensional reduction methods like Proper Orthogonal Decomposition (POD) or Auto-Encoders (AE) will first be used to compress to reduce the number of independent variables that are solved for every time step. However, this project focuses on DD-LSTM development.

The second step is to use LSTM to make the prediction of future time steps. In this step, I will apply LSTM to three test case of increased complexity. The first case is to use LSTM to do prediction on a simple advecting square wave. I am then using LSTM to do prediction on flow past a cylinder. In the final case, I will apply LSTM to predict the urban airflow in a small region of London. I hope to compare LSTM with Gaussian Process Regression (GPR).

The third part is to use the domain decomposition method to divide a large area into several small domains, using LSTM in each sub-domain and dealing with transporting airflow in adjacent sub-domains, see Figure 2. I hope to show an enhanced predictive ability of the resulting DD-LSTM over a more standard LSTM.

3 Literature Review and Proposed Approach

3.1 Non-Intrusive Reduce Order Modeling (NIROM)

Intrusive Reduce Order Modeling (IROM) and NIROM are two types of ROMs. NIROM does not depend on the source code of the HFM. Moreover, it can avoid some of the instability and non-linearity efficiency issues associated with IROM[27].

NIROM has been used to solve various flow dynamic problems. Both POD and AEs have been employed by multiple NIROMs in previous studies. Coefficients of the reduced basis functions can be recovered by the decoder part of the AE network[23] or through the POD's use of Singular Value Decomposition rotation matrices. Those coefficients will be used to recover the full model.

NIROMs can be divided into two stages, *offline* and *online*. The *offline* stage is responsible for dimensionality reduction and generates a reduced basis from the data. The *online* stage handles the prediction based on the reduced basis on the current time step, then recovers the solution to the problem in the full dimensional space[24].

Wang et al. proposed a non-intrusive POD reduced basis method for parametrised unsteady flows [23]. Xiao et al. developed a NIROM for predicting the turbulent air flows found within an urban environment [27]. Xiao et al. developed a Domain Decomposition Non-Intrusive Reduced Order Model (DDNIROM) for turbulent flows [26]. Xiao et al. present a DDNIROM for the Navier-Stokes equations to improve the capability of NIROM for complex flow problems over widely varying ranges of scales [25]. Xiao et al. modelled turbulent flow problem and an ocean gyre simulation [27;24]. In this project, similar NIROMs will be applied to the three test cases.

Here we are using data generated by the HFM. The NIROM is constructed by training a neural network using LSTM with data from the HFM. The data-driven approach, as well as being applied here, has also been used in a number of disciplines. Chinesta et al. used a data-driven method on linear and nonlinear elasticity [2]. Eggersmann et al. extended the Data-Driven formulation to the elasticity problems studied by Kirchdoerfer and Ortiz. They benefited from the data-driven method by generalizing the model to handle different scenarios [3]. The data-driven approach has also been used in computational mechanics and nonlinear elasticity [17;12].

Capuano et al. implemented the data-driven method on finite element formulations and demonstrated that the data-driven process could work with any machine learning and proved its effectiveness in reducing both the error and computational cost, see [1].

3.2 Long Short Term Memory Neural Network

LSTM is a Recurrent Neural Network (RNN) for overcoming the vanishing of gradients, and it is formed by an explicit memory cell and four gating units. Vanishing of gradients occurs when back-propagating errors across many time steps. But for LSTM, a self-connected recurrent edge for each memory cell will ensure the gradient will not vanish after passing across many times. This structure allows LSTMs to control the sequential information in the network [8;14].

LSTM is widely used in time series problems [20], such as stock price prediction, [15;11] diseases propagation, [19] and climate change prediction [28]. LSTMs are also commonly used in speech recognition [33;10;13], natural language processing [31;5], and classification problems [30;22;4].

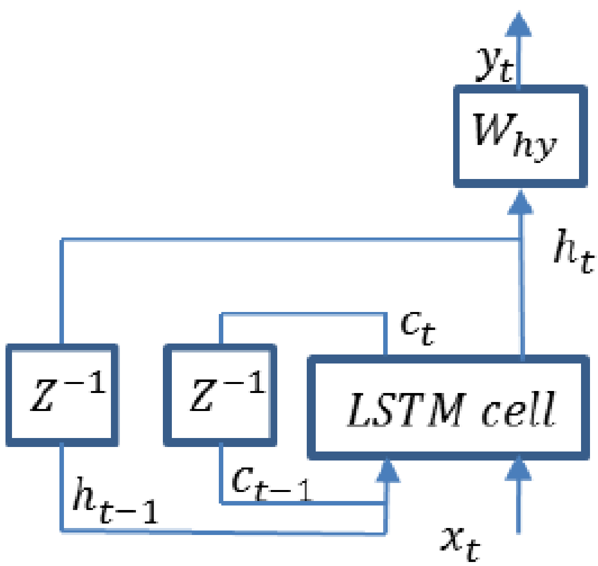

Figure 1: An LSTM Cell with One Recurrent Layer. Z^{-1} is the Time Delay Node

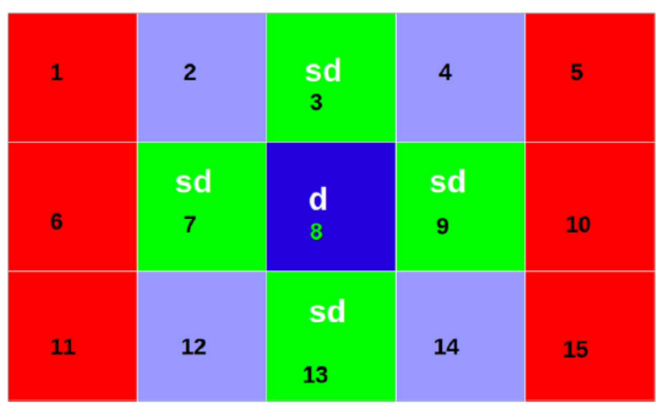

Figure 2: The figure shows a sub-domain d with its four neighbouring sub-domains [25]

As shown in Figure 1, the latest hidden-layer output is h_{t-1} , the latest memory activation is c_{t-1}. The latest memory activation is an input of the current LSTM cell [13].

The general LSTM use these four units: (Note: The output of an LSTM cells with l-th layer and time j is $h_j^l = o_j^l \tanh(c_j^l)$, b here is the bias term.)

(i) Memory units: store the temporal information.
The activation vector of Memory units:

$$c_j^l = f_j^l c_{j-1}^l + i_j^l \tanh(W_{xc}^l x_j^l + W_{hc}^l h_{j-1}^l + b_c^l) \tag{1}$$

(ii) Input gates: modulate the input activations into the cells.

The activation vector of input gates:

$$i_j^l = \sigma(W_{xi}^l x_j^l + W_{hi}^l h_{j-1}^l + W_{ci}^l c_{j-1}^l + b_i^l) \tag{2}$$

(iii) Output gates: modulate the output activations of the cells.
The activation vector of output gates:

$$o_j^l = \sigma(W_{xo}^l x_j^l + W_{ho}^l h_{j-1}^l + W_{co}^l c_j^l + b_o^l) \tag{3}$$

(iv) Forget gates: reset the cells memory.
The activation vector of forget gate:

$$f_j^l = \sigma(W_{xf}^l x_j^l + W_{hf}^l h_{j-1}^l + W_{cf}^l c_{j-1}^l + b_f^l) \tag{4}$$

In this project, LSTM is used to predict the compressed velocity time series [6]. For that, it is trained to learn the low dimensional dynamics of a fluid system. See Francisco and Maciej who used a single loss function for losses from both AE and LSTM. In their *offline* training process, the AE first takes N dimensional data and outputs the low dimensional representation. After the AE finishes the forward pass and constructs a batch of low-dimensional representations, the LSTM is trained with those representations. In their *online* prediction process, the trained AE retrieves the predictions from the LSTM in lower dimensions. The AE then uses the decoder in a forward pass to recover the predictions in the original dimension [6]. A similar structure will be used in this project, but we will introduce domain decomposition along with LSTM for implementation onto larger systems and call this method DD-LSTM. In addition, in this work, we decouple the compression (with either POD or AE) and the DD-LSTM *offline* training.

Gaussian Process Regression (GPR) has also be used in a similar way to LSTMs in previous studies, [32;9]. LSTM is suitable for dealing with essential events with longer intervals and delays in time series, and the GPR method has good adaptability and strong generality to process the complex nonlinear problems. In wind speed forecasting problem, both LSTM and GPR achieved higher forecasting accuracy than the conventional forecasting methods like Autoregressive Integrated Moving Average model and Back-Propagation Neural Network [9]. When forecasting high-dimensional chaotic system, the LSTM can better capture the nonlinear dynamics compared to GPR, while GPR has the better speed to obtain the same result than LSTM (20% percent better). As for computational complexity, the computational cost of the GPR-based approaches is significantly larger than LSTM. It was found that in short-term predictions, LSTM performs better. However, the prediction error increases faster for LSTM when increasing the chaoticity [21].

As for the setting of LSTM neural network, people tend to use not many hidden layers, single-layer LSTM is used in learning low-dimensional feature dynamics of fluid systems [6]. The number of layers is not mentioned in ocean gyre simulation and flow past cylinder studies, but it indicated the layer number is more than one [24]. Twenty hidden units and single hidden layers are used in the chaoticity study [21]. In classification problems, classification accuracy is related to the number of hidden layers [4]. In the mentioned prediction studies, people tend to use less hidden layer in LSTM network.

3.3 Domain Decomposition

DD is used to solve large problems in the frequency and time domains. The main feature of DD is decomposing large problems into several smaller ones. Instead of relying on iterative techniques in conventional DD, we choose to transform the original large matrix into one whose size is small enough to be manageable [16]. DD is often used with model reduction methods to conquer large problems [29].

Machine learning methods can be used to predict the geometric boundary for each domain. It reduces the number of eigenvalues before the iteration [7].

Multi-grid partitioning methods can be used to decomposed large finite element meshes into several sub-domains. Combining with automatic grid coarsening, unstructured meshes can also be decomposed. The sub-domain boundaries discourage the high crossing activity and poor conditioning but improve load balance [18]. In this project, the multi-grid partitioning method will be used to decompose the unstructured mesh.

Domain decomposition has been implemented to NIROM for modelling fluid flow and turbulent flow problems. Using Domain Decomposition methods allows one to construct local basis functions based on details of local flow solutions over each sub-domain. A Radial Basis Function (RBF) based NIROM is implemented, as several local basis functions are generated at first (by POD), RBF multi-dimensional interpolation method is then used to construct a set of hypersurfaces representing the local fluid dynamics over this sub-domain. When building the hypersurfaces for a given sub-domain, the solution in the surrounding sub-domains is taken into account by introducing their POD coefficients as inputs to the hypersurfaces of this particular sub-domain. DD-NIROM is suitable for hard problems which global Singular Value Decomposition may not decrease rapidly while its small sub-domains do [25].

The DD-LSTM method updates a sub-domain by taking the current value from itself and the future value from all its neighbours as the input, getting the output to update itself. In this case, as the input structure of every sub-domain is different because of the unstructured mesh, there will be the same number of LSTM models as the number of sub-domains, shown in Figure 2.

Xiao et al.[26] has used GPR and DD to form their NIROM. The approach developed in the current work is similar, but the LSTM replaces the GPR. In Xiao et al. the Reynolds stresses are used to decided how to weight each node of the mesh. The aim then is to balance the weighted sum in each sub-domain and minimize the communication or sum of the weights between the sub-domains. In this way, the sub-domains are equally balanced, in terms of the predictive ability, and the sub-domains form (as much as possible) dynamically isolated regions (sub-domains).

In our approach, the *offline* training process, first, divides up the domain into sub-domains, then forms the reduced order model the DD-LSTM is trained for each sub-domain separately. In the *online* process, the DD algorithm loops over all time steps, and forms the reduced order basis for each sub-domain separately. Then iterating over all the sub-domains the DD-LSTM is able to predict future time levels. Finally, project the reduced order solution is compared with the high fidelity solution.

The work flow for this project is shown in Figure 3:

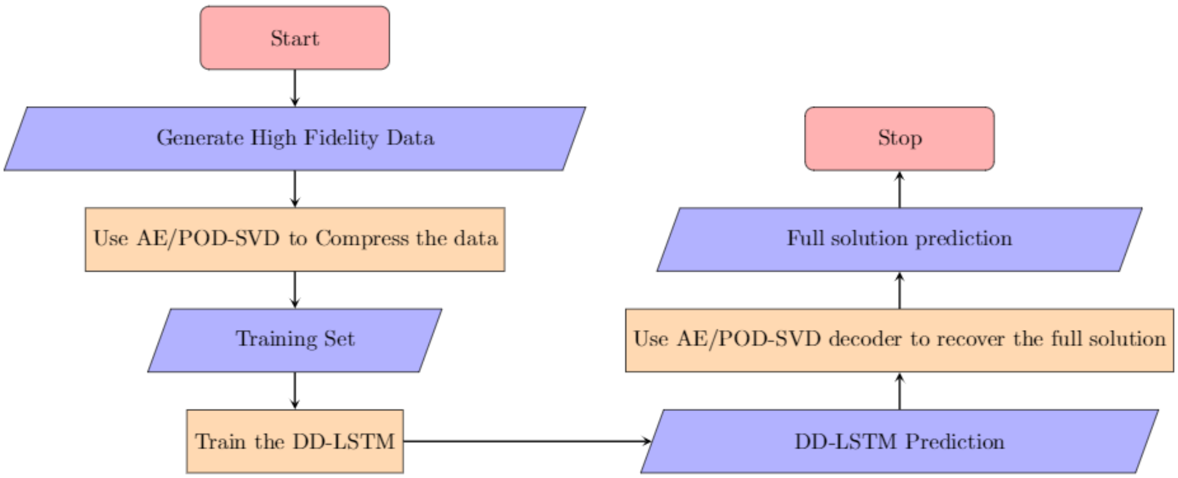

Figure 3: Algorithm describing the application of the DD-LSTM to an example problem.

4 Project Plan

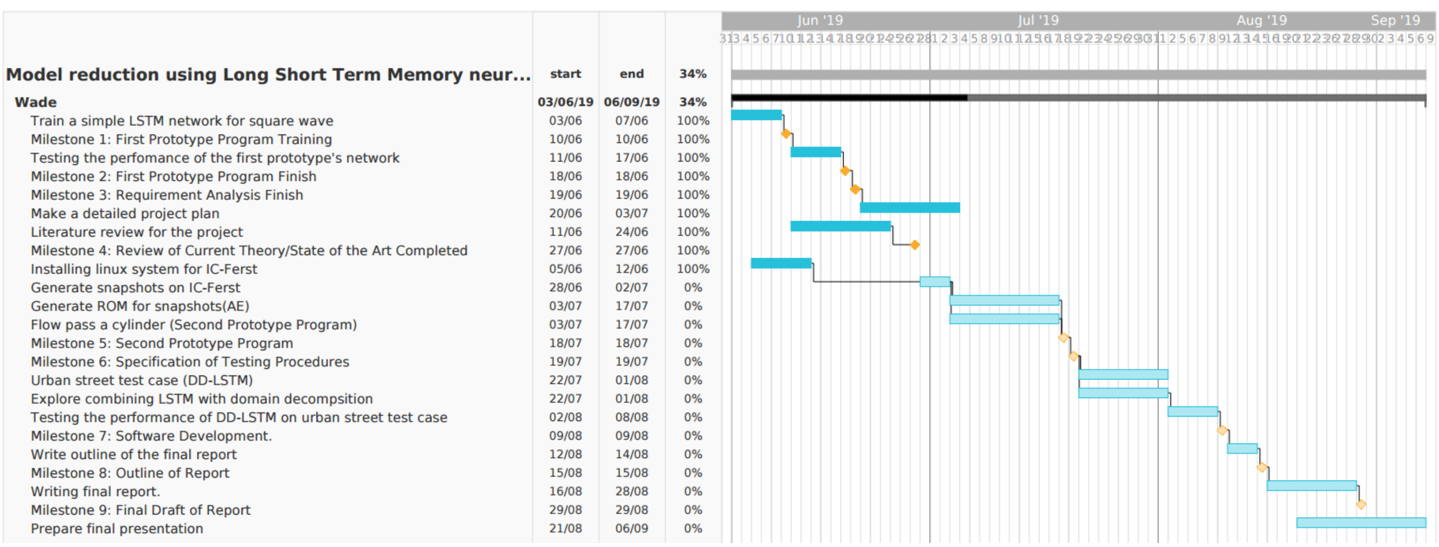

Figure 4: Gantt Chart of the Project Plan

References

[1] Capuano, G., and Rimoli, J. J. Smart finite elements: A novel machine learning application. *Computer Methods in Applied Mechanics and Engineering 345* (2019), 363–381.

[2] Chinesta, F., Ladeveze, P., Ibanez, R., Aguado, J. V., Abisset-Chavanne, E., and Cueto, E. Data-driven computational plasticity. *Procedia engineering 207* (2017), 209–214.

[3] Eggersmann, R., Kirchdoerfer, T., Reese, S., Stainier, L., and Ortiz, M. Model-free data-driven inelasticity. *Computer Methods in Applied Mechanics and Engineering* (2019).

[4] Ertugrul, A. M., and Karagoz, P. Movie genre classification from plot summaries using bidirectional lstm. In *2018 IEEE 12th International Conference on Semantic Computing (ICSC)* (2018), IEEE, pp. 248–251.

[5] Fahad, S. A., and Yahya, A. E. Inflectional review of deep learning on natural language processing. In *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)* (2018), IEEE, pp. 1–4.

[6] Gonzalez, F., and Balajewicz, M. Deep convolutional recurrent autoencoders for learning lowdimensional feature dynamics of fluid systems. *arXiv preprint arXiv:1808.01346* (2018).

[7] Heinlein, A., Klawonn, A., Lanser, M. H., and Weber, J. Machine learning in adaptive domain decomposition methods - predicting the geometric location of constraints. Technical report, Universität zu Köln, October 2018.

[8] Hochreiter, S., and Schmidhuber, J. Long short-term memory. *Neural computation 9*, 8 (1997), 1735–1780.

[9] Huang, Y., Liu, S., and Yang, L. Wind speed forecasting method using eemd and the combination forecasting method based on gpr and lstm. *Sustainability 10*, 10 (2018), 3693.

[10] Jo, J., Hwang, S., Lee, S., and Lee, Y. Multi-mode lstm network for energy-efficient speech recognition. In *2018 International SoC Design Conference (ISOCC)* (2018), IEEE, pp. 133–134.

[11] Kim, H. Y., and Won, C. H. Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models. *Expert Systems with Applications 103* (2018), 25–37.

[12] Kirchdoerfer, T., and Ortiz, M. Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering 304* (2016), 81–101.

[13] Li, J., Mohamed, A., Zweig, G., and Gong, Y. Lstm time and frequency recurrence for automatic speech recognition. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)* (2015), IEEE, pp. 187–191.

[14] Lipton, Z. C., Berkowitz, J., and Elkan, C. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019* (2015).

[15] Long, W., Lu, Z., and Cui, L. Deep learning-based feature engineering for stock price movement prediction. *Knowledge-Based Systems 164* (2019), 163–173.

[16] Mittra, R. A novel domain decomposition technique for solving very large problems in frequency and time domains. In *The Second European Conference on Antennas and Propagation, EuCAP 2007* (2007), IET, pp. 1–4.

[17] Nguyen, L. T. K., and Keip, M.-A. A data-driven approach to nonlinear elasticity. *Computers & Structures 194* (2018), 97–115.

[18] Pain, C., De Oliveira, C. E., and Goddard, A. A neural network graph partitioning procedure for grid-based domain decomposition. *International journal for numerical methods in engineering 44*, 5 (1999), 593–613.

[19] Park, H. D., Han, Y., and Choi, J. H. Frequency-aware attention based lstm networks for cardiovascular disease. In *2018 International Conference on Information and Communication Technology Convergence (ICTC)* (2018), IEEE, pp. 1503–1505.

[20] Shen, Z., Zhang, Y., Lu, J., Xu, J., and Xiao, G. A novel time series forecasting model with deep learning. *Neurocomputing* (2019).

[21] Vlachas, P. R., Byeon, W., Wan, Z. Y., Sapsis, T. P., and Koumoutsakos, P. Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 474*, 2213 (2018), 20170844.

[22] Wang, L., Xu, X., Dong, H., Gui, R., Yang, R., and Pu, F. Exploring convolutional lstm for polsar image classification. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium* (2018), IEEE, pp. 8452–8455.

[23] Wang, Q., Hesthaven, J. S., and Ray, D. Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. *Journal of Computational Physics 384* (2019), 289–307.

[24] Wang, Z., Xiao, D., Fang, F., Govindan, R., Pain, C. C., and Guo, Y. Model identification of reduced order fluid dynamics systems using deep learning. *International Journal for Numerical Methods in Fluids 86*, 4 (2018), 255–268.

[25] Xiao, D., Fang, F., Heaney, C., Navon, I., and Pain, C. A domain decomposition method for the non-intrusive reduced order modelling of fluid flow. *Computer Methods in Applied Mechanics and Engineering* (2019).

[26] Xiao, D., Heaney, C., Fang, F., Mottet, L., Hu, R., Bistrian, D., Aristodemou, E., Navon, I., and Pain, C. A domain decomposition non-intrusive reduced order model for turbulent flows. *Computers & Fluids 182* (2019), 15–27.

[27] Xiao, D., Heaney, C., Mottet, L., Fang, F., Lin, W., Navon, I., Guo, Y., Matar, O., Robins, A., and Pain, C. A reduced order model for turbulent flows in the urban environment using machine learning. *Building and Environment 148* (2019), 323–337.

[28] Yang, Y., Dong, J., Sun, X., Lima, E., Mu, Q., and Wang, X. A cfcc-lstm model for sea surface temperature prediction. *IEEE Geoscience and Remote Sensing Letters 15*, 2 (2017), 207–211.

[29] Yaoyao, L., Donglin, S., and Weimin, L. Higher order modes analysis of a hemp simulator using time-domain simulation and singular value decomposition. In *2018 International Applied Computational Electromagnetics Society Symposium-China (ACES)* (2018), IEEE, pp. 1–2.

[30] Yu, X., Xu, L., Ma, L., Chen, Z., and Yan, Y. Solar radio spectrum classification with lstm. In *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)* (2017), IEEE, pp. 519–524.

[31] Zhang, S., Liu, S., and Liu, M. Natural language inference using lstm model with sentence fusion. In *2017 36th Chinese Control Conference (CCC)* (2017), IEEE, pp. 11081–11085.

[32] Zhang, Y., Zhang, H., and Tian, Z. The application of gaussian process regression in state of health prediction of lithium ion batteries. In *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)* (2018), IEEE, pp. 515–519.

[33] Zhao, J., Mao, X., and Chen, L. Speech emotion recognition using deep 1d & 2d cnn lstm networks. *Biomedical Signal Processing and Control 47* (2019), 312–323.